

**AMENDMENTS TO THE CLAIMS**

Please amend the claims as follows.

1. (Currently Amended) A method for a system having distributed collaborating components, comprising:  
restricting direct interaction between distributed collaborating components by introducing an application-independent interface between distributed collaborating components; and  
invoking a service from the application-independent interface in order to enable interaction between distributed collaborating components, wherein invoking a service from the application-independent interface comprises sending a usage specification as an argument to the application-independent interface.
2. (Canceled)
3. (Previously Presented) The method of claim 1, wherein the application-independent interface has a capability to interpret the usage specification at runtime.
4. (Currently Amended) A method for a system having distributed collaborating components, comprising:  
restricting direct interaction between distributed collaborating components by introducing an application-independent interface between distributed collaborating components; and  
invoking a service from the application-independent interface in order to enable interaction between distributed collaborating components, wherein invoking a service from the application-independent interface comprises sending a logic execution specification as an argument to the application-independent interface.
5. (Original) The method of claim 4, wherein the application-independent interface has a capability to interpret the logic execution specification at runtime.
6. (Currently Amended) A method for a distributed system having a client and a server, comprising:

interposing a service layer between the client and the server, the service layer having a capability to interpret a specification sent as an argument from the client at runtime in order to enable interaction between the client and the server, wherein the specification is one selected from the group consisting of a usage specification and a logic execution specification; and routing correspondence between the client and server through the service layer.

7. (Previously Presented) The method of claim 6, wherein the usage specification comprises an attribute of an object on the server.
8. (Original) The method of claim 7, further comprising:  
the service layer fetching data from the object based on the specification.
9. (Original) The method of claim 8, wherein fetching data from the object comprises storing data fetched from the server in a proxy for the object.
10. (Original) The method of claim 7, further comprising the service layer updating data in the object based on the specification.
11. (Original) The method of claim 10, wherein updating data in the object comprises receiving data from the client and using data received from the client to modify the attribute of the object.
12. (Previously Presented) The method of claim 6, wherein the logic execution specification comprises logic for invoking a method of an object on the server.
13. (Previously Presented) The method of claim 12, wherein interpreting the logic execution specification comprises invoking the method of the object.
14. (Original) The method of claim 6, wherein interposing the service layer between the client and the server comprises separating specification of usage of an object on the server from implementation of the client.

15. (Original) The method of claim **14**, wherein interposing a service layer between the client and the server further comprises separating specification of logic for invoking a method of an object on a server from implementation of the client.
16. (Currently Amended) A computer-readable medium having recorded thereon instructions executable by a processor, the instructions for:  
receiving a specification as an argument from a client component; and  
interpreting the specification in order to enable interaction between the client component and a server component, wherein the specification is one selected from the group consisting of a usage specification and a logic execution specification.
17. (Previously Presented) The computer-readable medium of claim **16**, wherein the usage specification comprises a usage of an object in the server component.
18. (Original) The computer-readable medium of claim **17**, further comprising:  
instructions for fetching data from the object based on the usage.
19. (Original) The computer-readable medium of claim **17**, further comprising:  
instructions for updating data in the object based on the usage.
20. (Previously Presented) The computer-readable medium of claim **16**, wherein the logic execution specification comprises logic for invoking a method of an object on the server.
21. (Previously Presented) The computer-readable medium of claim **20**, wherein interpreting the logic execution specification comprises invoking the method of the object.
22. (Currently Amended) A distributed system, comprising:  
a client component;  
a server component having at least one object at runtime; and  
a service layer between the client and the server component, the service layer having a capability to interpret a specification of usage of the object sent as an argument at runtime.

23. (Original) The distributed system of claim 22, wherein the service layer further has a capability of interpret a specification of logic executions in the server component at runtime.
24. (Currently Amended) A distributed system, comprising:  
a service means for providing application-independent services and for interpreting a usage specification and a logic execution specification;  
a client component that sends the usage specification and a logic execution specification as an argument to the service means; and  
a server component that interacts with the service means in order to provide services to the client component.
25. (Canceled)
26. (New) The method of claim 1, wherein the usage specification comprises an attribute of an object on the server